7

Structures for lazy semantics

O. Bastonero, A. Pravato, S. Ronchi della Rocca
Università degli studi di Torino
Dipartimento di Informatica, C.so Svizzera 185, 10149 Torino, Italia
Email: bastone, pravato, ronchi@di.unito.it

Abstract

The paper explores different approaches for modeling the lazy λ-calculus, which is a paradigmatic language for studying the operational behaviour of programming languages, like Haskell, using a call-by-name and lazy evaluation mechanism. Two models for lazy λ-calculus in the coherence spaces setting are built. They give a new insight in the behaviour of the language since their local structures are different from the one of all existing models in the literature. In order to compare different models, a class of models for lazy λ-calculus is defined, namely, the lazy regular models class. All the models adequate for the lazy λ-calculus studied in the literature belong to this class.

Keywords

Lazy Lambda Calculus, Denotational Semantics, Coherent Spaces.

1 INTRODUCTION

Different structures have been explored for building models for the standard λ-calculus, like Scott domains, Engeler's algebras, DI-domains, filter models and coherence spaces. This richness has been very useful for grasping unexpected operational properties of λ-calculus. Indeed, it has been proved that there are λ-theories which can be modeled in Scott's domains but not in coherence spaces (Bastonero & Gouy to appear) and vice-versa (Bastonero 1996).

In this paper we are interested in studying structures for modeling the lazy λ-calculus. The lazy λ-calculus, formally introduced in Abramsky & Ong (1993), can be seen as a paradigm for programming languages, like Haskell (Hudak, Jones & Wadler 1992), using a call-by-name and a lazy evaluation mechanism. An evaluation mechanism is call-by-name when parameters are passed to a function before to be evaluated, moreover it is lazy when a procedure is evaluated only when its parameters are supplied. The syntax and the reduction rule of the lazy λ-calculus coincide with the syntax and reduction rule of the classical λ-calculus, but in this calculus a particular reduction strategy is introduced. This strategy leads to the concept of "value", namely a term on which the computation induced by the strategy stops. The set of values of the lazy λ-calculus contains all the abstraction terms, which formal-

Programming Concepts and Methods D. Gries, & W-P. de Roever (Eds.)

ize the notion of "procedure". Consequently, strictly speaking, all the models of the classical λ-calculus are models also for the lazy one, but we are interested in such models which are adequate with respect to the operational semantics modeling the call-by-name lazy evaluation mechanism. Models for lazy λ-calculus, adequate w.r.t. its operational semantics, have been studied in Longo (1983) and Abramsky & Ong (1993). The model of Longo ($\mathcal{M}_L$) is built in the Engeler's algebras setting, while the model of Abramsky and Ong ($\mathcal{M}_{AO}$) is built in the Scott's domains setting (they present it also as filter domain). These models have a very different local structure, namely, the theory of $\mathcal{M}_L$ is strictly included in the one of $\mathcal{M}_{AO}$, and neither the former nor the latter of the two models is fully abstract.

There is not, until now, a model for lazy λ-calculus built in the coherence spaces setting. Perhaps this depends on the fact that, since coherence spaces have been presented by Girard as a simplification of Scott domains, it would have been natural to mimic the Abramsky and Ong construction in the coherence spaces setting. But, this is not easy, since $\mathcal{M}_{AO}$ is based on the *lifting* constructor, which has not a natural corresponding construction in coherence spaces. And indeed, in order to mimic the lifting construction for modeling the lazy call-by-value evaluation in coherence spaces, Honsell and Lenisa defined a different notion of coherence space, the pointed coherence space (Honsell & Lenisa 1993).

Inside cartesian closed categories whose objects are cpo and morphisms are continuous functions, we define a class of models for lazy λ-calculus, the class of "lazy regular models", to which both the previous recalled models belong. All the models belonging to this class share important properties, like the approximation property. The definition of lazy regular models points out that a model for λ-calculus is adequate w.r.t. the lazy operational semantics if it is based on a domain $\mathcal{D}$ such that $(\mathcal{D} \Rightarrow \mathcal{D})$ is a retract of $\mathcal{D}$ (where $(\mathcal{D} \Rightarrow \mathcal{D})$ represents the function space from $\mathcal{D}$ to itself) and, if $\phi : \mathcal{D} \to (\mathcal{D} \Rightarrow \mathcal{D})$ and $\psi : (\mathcal{D} \Rightarrow \mathcal{D}) \to \mathcal{D}$ are the immersion, projection pair, $\psi(\bot_{(\mathcal{D} \Rightarrow \mathcal{D})}) \neq \bot_{\mathcal{D}}$. Let call *strict* a retraction of this kind. In fact the lifting construction is the easiest way of building a strict retraction in Scott domains. Consequently, in order to build a strict retraction in the coherence space setting, it is sufficient to have $\mathcal{D} \approx T(\mathcal{D} \Rightarrow \mathcal{D})$, where T is the interpretation of any additive or modal constructor. We exhibit two models of lazy λ-calculus, $\mathcal{M}_1$ and $\mathcal{M}_2$, built over the coherence space which is the minimal solution of the equation $\mathcal{D} \approx I\&(\mathcal{D} \Rightarrow_s \mathcal{D})$, where I is the coherence space with just one atom (interpreting the unity of the tensor product), & is the additive (cartesian) product, and $(\mathcal{D} \Rightarrow_s \mathcal{D})$ is the coherence space representing the stable functions from $\mathcal{D}$ to $\mathcal{D}$. In the paper we use a primitive constructor $(\cdot)_\emptyset$, isomorphic to $I\&\_$, that we call *pseudo-lifting*. Such a constructor can be viewed as a restricted power-domain construction.

It turns out that these two models give a new insight in the operational behaviour of the lazy λ-calculus. We prove that $\mathcal{M}_1$ has the same theory as $\mathcal{M}_L$, and we conjecture that $\mathcal{M}_2$ has the same theory as $\mathcal{M}_{AO}$. But the two new defined models have a local structure which is different from both the one of $\mathcal{M}_L$ and $\mathcal{M}_{AO}$. In particular, in all models all the unsolvable of infinite order are equated, but in $\mathcal{M}_L$ and $\mathcal{M}_{AO}$ such unsolvables are the greatest element, while in both $\mathcal{M}_1$ and $\mathcal{M}_2$ they are a maximal element, not comparable with any solvable term. An unsolvable of infinite order converges whatever sequence of arguments it is applied, while for a solvable term there is at least one sequence of arguments making it converging and at least one sequence making it diverging. Looking at the lazy λ-calculus as the tool for studying the "core" of lazy call-by-name real programming languages, an unsolvable of infinite order represents a program converging to an *undetermined* value, while a solvable term is such that it produces a particular, specified, value when applied to some sequence of arguments. Ong, in Ong (1992), interpreted the fact that the unsolvable of infinite order are the top in $\mathcal{M}_L$ and $\mathcal{M}_{AO}$, by the fact that they represent an *over-specified* value. This is reasonable, but the two models we exhibit show that it is possible to have a more refined interpretation, when undetermined and determined values are interpreted in not comparable points.

As far as the full-abstraction problem is concerned, no one of the two models is fully-abstract: indeed, to $\mathcal{M}_2$ the counter-example used in Theorem 8.1.1 of Abramsky & Ong (1993) can be applied, inducing that $\mathcal{M}_1$ too is not fully-abstract.

An example of model for lazy (but this time *call-by-value*) computation built using a modal construction in the category of coherence spaces and linear functions, can be found in Pravato, Ronchi della Rocca & Roversi (1995).

2 LAZY MODELS

In this section we will define the lazy λ-calculus and its models. Moreover, we will introduce a class of models, particularly suitable for our purposes.

Let Λ be the set of terms of the pure (i.e. without constants) λ-calculus built out from a denumerable set *Var* of variables. I.e., terms of Λ are generated by the following grammar: $M ::= x \mid MM \mid \lambda x.M$ where x ranges over *Var*. Let $\Lambda^o \subseteq \Lambda$ be the set of closed terms.

Let $\Rightarrow_\beta$ denote the contextual transitive and reflexive closure of the β-reduction rule $\rightarrow_\beta$ on terms, defined as: $(\lambda x.M)N \rightarrow_\beta M[N/x]$. Let denote with $=_\beta$ the symmetric contextual transitive and reflexive closure of $\rightarrow_\beta$.

Definition 1 *Let Val* $= \{\lambda x.M \mid M \in \Lambda\} \subseteq \Lambda$.
i) The notion of lazy convergence *for closed terms of the λ-calculus is defined through a logical system proving judgements of the shape: $M \Downarrow_{lazy} P$ where $M \in \Lambda^o$ and $P \in Val$.*
The rules of the system hare the following:

$$\frac{}{\lambda x.M \Downarrow_{lazy} \lambda x.M}(abs) \qquad\qquad \frac{M \Downarrow_{lazy} \lambda x.P \qquad P[N/x] \Downarrow_{lazy} Q}{MN \Downarrow_{lazy} Q}(app)$$

Let $M \Downarrow_{lazy}$ (read: M is valuable *or M* converges*) be an abbreviation for $\exists P.M \Downarrow_{lazy} P$, and $M \Uparrow_{lazy}$ (M* diverges*) be an abbreviation for $\nexists P.M \Downarrow_{lazy} P$. Notice that this notion of convergence is a particular β-reduction strategy, namely if $M \Downarrow_{lazy} P$, then $M \Rightarrow_\beta P$.*

ii) The lazy operational preorder *on* Λ *is defined as follows: for all* $M, N \in \Lambda$
$M \sqsubseteq_{lazy} N \Leftrightarrow (\forall C[.] \,.\, C[M], C[N] \in \Lambda^o \,.\, C[M] \Downarrow_{lazy} \Rightarrow C[N] \Downarrow_{lazy})$
Let $\approx_{lazy}$ *be the equivalence induced by* $\sqsubseteq_{lazy}$.

In order to model the lazy convergence of the λ-calculus, a semantic account of the notion of valuable terms must be given. This can be done by enriching the well–known set–theoretical definition of λ-calculus model in Hindley-Longo style, see Hindley & Longo (1980), by explicitly introducing a subset V of the interpretation domain, where all valuable terms have to be interpreted.

Definition 2 *A* model for the lazy λ-calculus *is a 4-tuple* $\mathcal{M} = (D, \bullet, V, [\![\cdot]\!])$ *where: D together with $\bullet$ is an applicative structure, i.e., it is a set with at least two elements equipped with a binary operation* $\bullet : D^2 \to D$, $V \subseteq D$ *is the set of* semantic values, $[\![\cdot]\!] : \Lambda \to \mathbf{Env} \to D$ *is the* interpretation function, *where* $\mathbf{Env}$ *is the set of* environments, *(i.e. maps* $\rho : Var \to D$*), such that* $(D, [\![\cdot]\!])$ *is a model for the pure λ-calculus, and* $(M \in Val \implies \forall \rho \in \mathbf{Env} \,.\, [\![M]\!]_\rho \in V)$.

The semantic equivalence of terms in $\mathcal{M}$ *is defined as usual: given* $M, N \in \Lambda$,
$\mathcal{M} \models M = N \iff (\forall \rho \in \mathbf{Env} \,.\, [\![M]\!]_\rho = [\![N]\!]_\rho)$.

Remember that a model $\mathcal{M}$ for a given λ-calculus is *adequate* w.r.t. a $\approx$ operational semantics if and only if $\mathcal{M} \models M = N$ implies $M \approx N$. The following proposition gives an equivalent notion of equivalence for lazy models, based on the notion of semantics values.

Proposition 3 *A model* $\mathcal{M} = (D, \bullet, V, [\![\cdot]\!])$ *for the lazy λ-calculus is adequate for the lazy operational equivalence* $\approx_{lazy}$ *if:* $\forall M \in \Lambda^o \,.\, ([\![M]\!] \in V \implies M \Downarrow_{lazy})$.

Remark 4 In Abramsky & Ong (1993) we find a general definition of a model of the lazy λ-calculus in a different but equivalent style, starting from the basic notion of applicative structure equipped with a partial evaluation function. They require explicitly that a model for the lazy λ-calculus be adequate w.r.t. $\approx_{lazy}$. If Definition 2 was modified by adding the condition of Proposition 3, so requiring explicitly the adequacy, then the resulting definition would be equivalent to the one given in Abramsky & Ong (1993). We chosed to have adequacy as an additional condition for sake of uniformity with the standard λ-calculus treatment. And indeed, there is just one definition of what is a λ-calculus model, but a given model may be adequate or not w.r.t. a given operational semantics. For example, the standard D_∞ Scott's model is adequate w.r.t. the operational semantics induced by a call-by-name reduction machine (whose values are the head-normal-forms), but it is not adequate to the one induced by the normalizing machine (whose values are all the normal-forms). An example of a model adequate w.r.t. this last operational semantics is the model described in Coppo, Dezani-Ciancaglini & Zacchi (1987).

2.1 Lazy regular models

For our purposes, we need to consider only a particular class of models for the lazy λ-calculus, the *lazy-regular*-models, based on the notion of domains. All the models we are interested in belong to this class. We will prove that all lazy-regular models are adequate w.r.t. the given operational semantics. First of all, let recall the definition of *regular* model for λ-calculus as given in Bastonero & Gouy (to appear).

Let $\mathbf{C}$ be a closed cartesian category (c.c.c.) with enough points. Let $\Rightarrow: \mathbf{C} \times \mathbf{C} \rightarrow \mathbf{C}$ be the bifunctor such that, for all objects A, B, the object $A \Rightarrow B$ internalizes the set $\mathbf{Hom_C}(A, B)$ of morphisms from the object A to the object B, through the operation $\Lambda_{A,B,C} : \mathbf{Hom_C}(A \times B, C) \rightarrow \mathbf{Hom_C}(A, (B \Rightarrow C))$ and the *evaluation morphism*: $ev_{B,C} : (B \Rightarrow C) \times B \rightarrow C$, for each C (see Asperti & Longo (1991) for details).

Definition 5 *A regular category* $\mathbf{C}$ *is a c.c.c. such that: every object* A *of* $\mathbf{C}$ *is a pointed complete partial order w.r.t. the order relation* $\leq_A$, *where* $\perp_A$ *denotes the minimum point; the morphisms of* $\mathbf{C}$ *are continuous functions.*

For each morphism f from an object A to an object B, $\Lambda(f)$ is a shortening for $\Lambda_{\mathbb{I},A,B}(f \circ \lambda_A)$ (where λ_A is the isomorphism between A and $\mathbb{I} \times A$, taking $\mathbb{I}$ as the unity of the cartesian product $\times$) which is the point in $A \Rightarrow B$ corresponding to f. Conversely, Λ^{-1} will be used as a shortening for $\Lambda^{-1}_{\mathbb{I},A,B}$.

Definition 6 *Let* $\mathcal{D}$ *be a reflexive object of a regular category, namely, an object such that* $\mathcal{D} \rhd (\mathcal{D} \Rightarrow \mathcal{D})$ *through the retraction pair* (ϕ, ψ). $\mathcal{D}$ *is an applicative structure once defined a binary operator* $\bullet$ *in the following way:*

$\forall d_1, d_2 \in \mathcal{D} \,.\, d_1 \bullet d_2 = (\Lambda^{-1}(\phi(d_1)))(d_2)$.

Then $\mathcal{M} = (D, \bullet, [\![\cdot]\!])$ *is a* regular model *of* λ*-calculus, where the interpretation function is defined as follows:*

- $[\![x]\!]_\rho = \rho(x)$;
- $[\![MN]\!]_\rho = (\Lambda^{-1}(\phi([\![M]\!]_\rho)))([\![N]\!]_\rho)$;
- $[\![\lambda x.M]\!]_\rho = \psi(\Lambda(d \in \mathcal{D} \mapsto [\![M]\!]_{\rho[d/x]}))$.

Definition 7 *Let* $\mathcal{M} = (D, \bullet, [\![\cdot]\!])$ *be a regular model.*

i) The set of projections *of* $\mathcal{M}$ *is the set of points* $p \in \mathcal{D}$ *such that:* $p \leq_{\mathcal{D}} \psi(\Lambda(d \in \mathcal{D} \mapsto d))$ *and* $p = \psi(\Lambda(d \in \mathcal{D} \mapsto p \bullet (p \bullet d)))$. *Let us notice that for every projection* p *we have:* $\forall d \in \mathcal{D} \,.\, p \bullet d = p \bullet (p \bullet d)$.

ii) $\mathcal{M}$ *is* stratified *if there exists an increasing sequence of projections* $p_0 \leq_{\mathcal{D}} p_1 \leq_{\mathcal{D}} p_2 \leq_{\mathcal{D}} \cdots$ *in* $\mathcal{D}$ *such that:*

- $\bigsqcup_n p_n = \psi(\Lambda(d \in \mathcal{D} \mapsto d))$,
- $p_{n+1} \bullet d_1 \bullet d_2 = p_n \bullet (d_1 \bullet (p_n \bullet d_2))$ *for all* $d_1, d_2 \in \mathcal{D}$,
- $p_0 \bullet d = \perp_{\mathcal{D}}$ *for all* $d \in \mathcal{D}$.

Definition 8 *A* lazy regular model $\mathcal{M} = (\mathcal{D}, \bullet, \mathcal{V}, [\![\cdot]\!])$ *is a model for the lazy* λ-*calculus (as in Definition 2) where:*
i) $(\mathcal{D}, \bullet, [\![\cdot]\!])$ *is a stratified regular model;*
ii) $\psi : (\mathcal{D} \Rightarrow \mathcal{D}) \rightarrow \mathcal{D}$ *is not strict, i.e.,* $\psi(\bot_{(\mathcal{D}\Rightarrow\mathcal{D})}) \neq \bot_{\mathcal{D}}$,
iii)$\bot_{\mathcal{D}} \notin \mathcal{V}$.

Since every regular model is built over a partial order, we can now extend the notion of adequacy. Namely a lazy regular model $\mathcal{M} = (\mathcal{D}, \bullet, \mathcal{V}, [\![\cdot]\!])$ is adequate for $\sqsubseteq_{lazy}$ if and only if $\mathcal{M} \models M \sqsubseteq N$ implies $M \sqsubseteq_{lazy} N$, where $\mathcal{M} \models M \sqsubseteq N$ means $\forall \rho.[\![M]\!]_\rho \leq_{\mathcal{D}} [\![N]\!]_\rho$.

Theorem 9 *Every lazy regular model is adequate for the lazy operational preorder* $\sqsubseteq_{lazy}$.

Proof. In the following section we will prove that every lazy regular model enjoys an approximation property w.r.t. a suitable notion of approximants. The adequacy is a direct consequence of this fact. □

3 APPROXIMATION IN LAZY REGULAR MODELS

In this section we establish that the notion of *Lévy–Longo preorder*, introduced in Longo (1983) by adapting an idea of Lévy (Lévy 1975), is a key one for speaking about the theories of lazy regular models. Namely, the Lévy–Longo preorder is included in the local theory of every lazy regular model. The proof of this last point is an immediate consequence of the fact that all the lazy regular models share an approximation property w.r.t. the same notion of approximant, which is an adaptation of the one introduced by Hyland (Hyland 1976) and Wadsworth (Wadsworth 1978) for analyzing Scott's D_∞ model. The approximation property says that the interpretation of a term is the supremum of the interpretations of a set of normal forms in an extended language (its *approximants*).

3.1 Approximation Theorem

Let the $\lambda(\Omega)$-calculus be the following extension of the pure λ-calculus obtained by adjoining to set of variable the constant Ω. The rewriting rules are the β-rule and also the Ω-rule defined by: $(\Omega) \qquad \Omega M \rightarrow_\Omega \Omega$.
Every lazy regular model $(\mathcal{D}, \bullet, \mathcal{V}, [\![\cdot]\!])$ can be extended to $\Lambda(\Omega)$ posing $[\![\Omega]\!]_\rho = \bot_{\mathcal{D}}$.

Definition 10 *Let* $M \in \Lambda$.
$A \in \Lambda(\Omega)$ *is an* approximation *of* M*, and we write* $A\, app\, M$*, if* A *is obtained from* M *by substituting* Ω *for some subterm of* M.
Let $\mathcal{N}_{\beta\Omega} \subseteq \Lambda(\Omega)$ *be the set of* $\beta\Omega$*-normal forms. The set of* approximants *of* M *is the set* $\mathcal{A}(M) = \{A \in \mathcal{N}_{\beta\Omega} \mid \exists M'.(M \Rightarrow_\beta M') \wedge (A\, app\, M')\}$.

It is immediate to verify that the terms of $\mathcal{N}_{\beta\Omega}$ have either the shape $\lambda x_1 \ldots x_n.\Omega$ or $\lambda x_1 \ldots x_n.xA_1 \ldots A_p$, where $A_i \in \mathcal{N}_{\beta\Omega}$ (for $p \geq 0, n \geq 0$).

For lazy regular models an approximation theorem holds. It can be proved by adapting the proof given in Ong (1992) (theorem 4.8) for a class of models of lazy λ-calculus which is properly included in our class.

Theorem 11 (Approximation Theorem) *Let* $\mathcal{M} = (\mathcal{D}, \bullet, \mathcal{V}, [\![\cdot]\!])$ *be a lazy regular model. For all* $\rho \in$ **Env** *and* $M \in \Lambda$, $[\![\mathcal{A}(M)]\!]_\rho$ *is directed in* $\mathcal{D}$ *and* $[\![M]\!]_\rho = \bigsqcup\{[\![A]\!]_\rho \mid A \in \mathcal{A}(M)\}$.

3.2 Lazy preorders

In order to introduce some notions of Lazy preorders, we need to recall the notion of order of an unsolvable.

An unsolvable term M is of *order 0* if $\not\exists P.M =_\beta \lambda x.P$. An unsolvable term M is *of order n* ($n > 0$) if n is the maximum m such that $\exists P.M =_\beta \lambda x_1 \ldots x_m.P$. If such an n does not exist, then M is *unsolvable of infinite order.*

Let us recall the Lévy-Longo preorder ($\sqsubseteq_{LT}$) given in Longo (1983) and studied in Ong (1992), and the $\sqsubseteq_{PSE}$ preorder, defined in Ong (1992). Moreover we need to define a new preorder between terms, the stable lazy one, denoted by $\sqsubseteq_{LS}$, for studying the local structure of the models we will introduce later.

Definition 12 $M \sqsubseteq_i N$ *if and only if* $\forall k \geq 0.M \sqsubseteq_i^k N$ *where* $i \in \{LT, LS, PSE\}$ *and* $M \sqsubseteq_i^k N$ *is so defined:*

- $M \sqsubseteq_i^0 N$ *always holds;*
- $M \sqsubseteq_{LT}^{k+1} N$ *if and only if:*
  (i) M *is unsolvable of order* n *and* $\exists P \in \Lambda$ *s.t.* $N =_\beta \lambda x_1 \ldots x_n.P$, *or*
  (ii) M *is unsolvable of infinite order and* N *is unsolvable of infinite order, or*
  (iii) $M =_\beta \lambda x_1 \ldots x_n.yM_1 \ldots M_p$ *and* $N =_\beta \lambda x_1 \ldots x_n.yN_1 \ldots N_p$ *and* $\forall 1 \leq i \leq p.M_i \sqsubseteq_{LT}^k N_i$;
- $M \sqsubseteq_{PSE}^{k+1} N$ *if and only if:*
  (i) M *is unsolvable of order* n *and* $\exists P \in \Lambda$ *s.t.* $N =_\beta \lambda x_1 \ldots x_n.P$, *or*
  (ii) N *is unsolvable of infinite order, or*
  (iii) $M =_\beta \lambda x_1 \ldots x_n.yM_1 \ldots M_p$ *and* $N =_\beta \lambda x_1 \ldots x_{n+q}.yN_1 \ldots N_{p+q}$ *and* $\forall 1 \leq i \leq p.M_i \sqsubseteq_{PSE}^k N_i$ *and* $\forall 1 \leq j \leq q.x_{n+j} \sqsubseteq_{PSE}^k N_{p+j}$.
- $M \sqsubseteq_{LS}^{k+1} N$ *if and only if:*
  (i) M *is unsolvable of order* n *and* $\exists P \in \Lambda$ *s.t.* $N =_\beta \lambda x_1 \ldots x_n.P$, *or*
  (ii) M *is unsolvable of infinite order and* N *is unsolvable of infinite order, or*
  (iii) $M =_\beta \lambda x_1 \ldots x_n.yM_1 \ldots M_p$ *and* $N =_\beta \lambda x_1 \ldots x_{n+q}.yN_1 \ldots N_{p+q}$ *and* $\forall 1 \leq i \leq p.M_i \sqsubseteq_{LS}^k N_i$ *and* $\forall 1 \leq j \leq q.x_{n+j} \sqsubseteq_{LS}^k N_{p+j}$.

All the three preorders share the property that they equate all the unsolvables of infinite order. By the way, such unsolvables are the greatest element in $\sqsubseteq_{PSE}$, while in both $\sqsubseteq_{LT}$ and $\sqsubseteq_{LS}$ they are maximal elements, not the top. As far as solvable terms are concerned, $M \Rightarrow_\eta N$ implies $N \sqsubseteq_i M$ ($i \in \{LS, PSE\}$), while this implication does not hold for $\sqsubseteq_{LT}$.

Proposition 13 $\sqsubseteq_{LT} \subset \sqsubseteq_{LS} \subset \sqsubseteq_{PSE}$ *and they induce the same equivalence relation.*

Theorem 14 *Let $\mathcal{M}$ be a lazy regular model and let $M, N \in \Lambda$.*
$M \sqsubseteq_{LT} N \Longrightarrow (\mathcal{M} \models M \sqsubseteq N)$.

Proof. (Sketch) Using the characterisation of approximants, we show that if $M \sqsubseteq_{LT} N$ then $\mathcal{A}(M) \subseteq \mathcal{A}(N)$. Then the thesis follows from the approximation theorem. $\square$

Let $\mathcal{M} = (\mathcal{D}, \bullet, \mathcal{V}, [\![\cdot]\!])$ be a lazy regular model. We denote with $\mathcal{M} \models \eta^-$ the fact that for all $d \in \mathcal{D}$, $\mathcal{M} \models d \sqsubseteq \lambda x.dx$.

Theorem 15 *Let $\mathcal{M}$ be a lazy regular model such that $\mathcal{M} \models \eta^-$. Let $M, N \in \Lambda$.*
$M \sqsubseteq_{LS} N \Longrightarrow (\mathcal{M} \models M \sqsubseteq N)$.

Proof. (Sketch) Let $M \sqsubseteq_{LS} N$, then $M \sqsubseteq_{LT} N'$ with $N \Rightarrow_\eta N'$. By theorem 14, $\mathcal{M} \models M \sqsubseteq N'$ and, by induction, using the fact that $\mathcal{M} \models \eta^-$ and the continuity, $\mathcal{M} \models M \sqsubseteq N$. $\square$

4 LAZY REGULARITY IN COHERENCE SPACES

Let us consider the category **Stab**, whose objects are *coherence spaces* and whose morphisms are *stable functions*, which can be easily proved to be regular. For what concerns the notions relative to coherence spaces and stability, we refer the reader to Girard, Lafont & Taylor (1989).

4.1 The models $\mathcal{M}_1$ and $\mathcal{M}_2$

If D is a coherence space, let us denote with $|D|$ its set of atoms (ranged over by α, β, γ,...), and by $comp_D \subseteq |D| \times |D|$ the compatibility relation. Elements of D are subsets of compatible atoms, and will be denoted by $a, b, c, \ldots$. The bottom elements is the empty set ($\emptyset$). We also denote by $(D_1 \Rightarrow_s D_2)$ the coherence space whose elements are the traces of the stable functions from D_1 to D_2. Let us recall that the trace of a stable function f from D_1 to D_2 is the set $tr(f) \subseteq (D_1)_{\text{fin}} \times |D_2|$ (where $(D_1)_{\text{fin}}$ denotes the set of all the finite elements of D_1) defined as:
$tr(f) = \{(a_0, \beta) \mid \forall a.(\beta \in f(a) \Rightarrow a_0 \subseteq a)\}$
where, if $(a, \beta), (a', \beta') \in \mathcal{F}$ and $a \cup a' \in D_1$ then both $comp_{D_2}(\beta, \beta')$ and if

$\beta = \beta'$ then $a = a'$. A stable function f is completely determined by its trace $tr(f)$ in the following way: $f(a) = \{\beta \mid \exists a_0 \subseteq a.(a_0, \beta) \in tr(f)\}$.

Now, in order to define lazy regular models in **Stab** we introduce an endofunctor $(\cdot)_\emptyset$.

Definition 16 *Let us define the* pseudo-lifting*functor:* $(\cdot)_\emptyset$: **Stab** $\to$ **Stab** *as:*

- *if D is a coherence space, then $D_\emptyset$ is the coherence space whose set of atoms is: $|D_\emptyset| = \{a \mid a \subseteq \mathcal{P}_{fin}(|D|), \#a \leq 1\}$ and whose compatibility relation is defined by: $\forall a, b \in |D_\emptyset|.(a, b) \in comp_{D_\emptyset} \Leftrightarrow a \cup b \in D$ ($\#a$ denotes the cardinality of the set a),*
- *if f is a stable function from D_1 to D_2, then $f_\emptyset$ is the stable function from $(D_1)_\emptyset$ to $(D_2)_\emptyset$ whose trace is: $tr(f_\emptyset) = \{(\{\emptyset\}, \emptyset)\} \cup \{(\tilde{a}, \{\alpha\}) \mid (a, \alpha) \in tr(f)\}$ where, if $a = \{\alpha_1, \ldots, \alpha_n\}$ with $n > 0$, then $\tilde{a} = \{\{\alpha_1\}, \ldots, \{\alpha_n\}\}$, while $\tilde{\emptyset} = \{\emptyset\}$.*

Note that the atom $\emptyset$ is compatible with all other atoms of $D_\emptyset$, according to the definition above. Moreover, for every coherence space D, $D_\emptyset \rhd D$ through both the retraction pair (out^1_D, in_D) and (out^2_D, in_D), where:
- $in_D : D \to D_\emptyset$ is defined as: $in_D(a) = \{\{\alpha\} \mid \alpha \in a\} \cup \{\emptyset\}$, whose trace is: $tr(in_D) = \{(\{\alpha\}, \{\alpha\}) \mid \alpha \in |D|\} \cup \{(\emptyset, \emptyset)\}$.
- for $i \in \{1, 2\}$, $out^i_D : D_\emptyset \to D$, are defined as: $out^1_D(b) = \{\alpha \mid \{\alpha\} \in b\}$, whose trace is $tr(out^1_D) = \{(\{\{\alpha\}\}, \alpha) \mid \alpha \in |D|\}$, $out^2_D(b) = \{\alpha \mid \{\alpha\} \in b \text{ and } \emptyset \in b\}$, whose trace is $tr(out^2_D) = \{(\{\{\alpha\}, \emptyset\}, \alpha) \mid \alpha \in |D|\}$. Note that the behaviour of out^1_D and out^2_D is the same when they are applied to elements in the range of in_D.

Remark 17 It is routine to check that $D_\emptyset$ is isomorphic to $I\&D$, where I is the coherence space with just one atom.

We now define two lazy regular models. Both are based on the coherence space $\mathcal{D}$, minimum solution of the domain equation $D \approx (D \Rightarrow_s D)_\emptyset$. Such a solution $\mathcal{D}$ can be obtained by the inverse limit construction, namely, $\mathcal{D} \approx \lim_{\leftarrow} \langle D_n, j_n \rangle$, where $|D_0| = \emptyset$ and $D_{n+1} = (D_n \Rightarrow_s D_n)_\emptyset$, and the initial immersion-projection pair is (i_0, j_0), where $i_0 = x \in D_0 \mapsto \emptyset$, and $j_0 = x \in D_1 \mapsto \emptyset$. Let (F, G) be the resulting isomorphism pair. Using the fact that $(\mathcal{D} \Rightarrow_s \mathcal{D})_\emptyset \rhd (\mathcal{D} \Rightarrow_s \mathcal{D})$, through the retraction pairs $(out^1_{\mathcal{D}\Rightarrow_s\mathcal{D}}, in_{\mathcal{D}\Rightarrow_s\mathcal{D}})$ and $(out^2_{\mathcal{D}\Rightarrow_s\mathcal{D}}, in_{\mathcal{D}\Rightarrow_s\mathcal{D}})$, we define the retraction $D \rhd (D \Rightarrow_s D)$ through the retraction pairs (ϕ^i, ψ), for $i \in \{1, 2\}$, where $\phi^i = out^i_{\mathcal{D}\Rightarrow_s\mathcal{D}} \circ F$ and $\psi = G \circ in_{\mathcal{D}\Rightarrow_s\mathcal{D}}$. Hence, we obtain two different regular models. Based on these regular models are the lazy regular models $\mathcal{M}_1$ and $\mathcal{M}_2$ that we will define after the following remark:

Remark 18 Since the isomorphism pair (F, G) defined above, we can characterize the set of atoms $|\mathcal{D}|$ as it was the set of atoms $|(\mathcal{D} \Rightarrow_s \mathcal{D})_\emptyset|$. Furthermore, reasoning modulo isomorphism, if $\gamma \in |\mathcal{D}|$, then $\gamma = \emptyset$, or else $\gamma = \{(a, \alpha)\}$, where $a \in \mathcal{D}$, a finite and $\alpha \in |\mathcal{D}|$.

Let $\mathcal{M}_1 = (\mathcal{D}, \bullet_1, \mathcal{V}, [\![\cdot]\!]^1)$ and $\mathcal{M}_2 = (\mathcal{D}, \bullet_2, \mathcal{V}, [\![\cdot]\!]^2)$ be defined as follows:

a) $\mathcal{V} = \{a \in \mathcal{D} \mid \emptyset \in a\}$,
b) $d_1 \bullet_1 d_2 = \{\alpha \mid \exists a \subseteq d_2.\{(a, \alpha)\} \in d_1\}$,
c) $d_1 \bullet_2 d_2 = \{\alpha \mid \exists a \subseteq d_2.\{\emptyset, \{(a, \alpha)\}\} \subseteq d_1\}$,
d) the interpretation functions $[\![\cdot]\!]^1$ and $[\![\cdot]\!]^2$ are obtained by instantiating, in the definition given after Definition 6, ϕ by ϕ^1 and ϕ^2, and considering $\Lambda(d \in \mathcal{D} \mapsto [\![M]\!]^i_{\rho[d/x]})$ as a notation for the trace:
$\{(a, \alpha) \mid a \text{ is the minimal finite } a' \in \mathcal{D} \text{ s.t. } \alpha \in [\![M]\!]^i_{\rho[a'/x]}\}$.

Proposition 19 *$\mathcal{M}_1$ and $\mathcal{M}_2$ are lazy regular models.*

Proof. We must prove just that $\mathcal{M}_1$ and $\mathcal{M}_2$ satisfy Definition 8.

First of all, they must be models for the lazy λ-calculus. In order to prove this, only point 2 of Definition 2 needs a verification. This is immediate since if $M \in \mathit{Val}$ then, for each ρ, $[\![M]\!]^i_\rho = \psi(\Lambda(f))$, for a given $f : \mathcal{D} \to \mathcal{D}$. Since $\psi = G \circ in_{\mathcal{D} \Rightarrow_s \mathcal{D}}$ and $\emptyset \in in_{\mathcal{D} \Rightarrow_s \mathcal{D}}(d)$ for each $d \in (\mathcal{D} \Rightarrow_s \mathcal{D})$, speaking modulo the isomorphism G, $\emptyset \in [\![M]\!]^i_\rho$.

Secondly, we prove that they are stratified. We make the prove only for $\mathcal{M}_2$, since for $\mathcal{M}_1$ the proof is similar, but simpler.

First of all, let us define the rank of an atom in $|\mathcal{D}|$ as the following integer:
$\text{rg}(\emptyset) = 0$, $\text{rg}(\{(a, \alpha)\}) = \max\{\max\{\text{rg}(\beta) \mid \beta \in a\}, \text{rg}(\alpha)\} + 1$.

If $a \in \mathcal{D}$ is finite, then $\text{rg}(a)$ denotes $\max\{\text{rg}(\beta) \mid \beta \in a\}$. We denote by $|\mathcal{D}|_n$ the set of atoms of $\mathcal{D}$ whose rank is smaller of equal to n.

The interpretation of the identity term $\lambda x.x$ is given by, (omitting G, up to the isomorphism), $\{\{(\{\alpha\}, \alpha)\} \mid \alpha \in |\mathcal{D}|\} \cup \{\emptyset\}$.

We take as p_n, for $n \geq 0$, the following points:
$p_n = \{\gamma \in [\![\lambda x.x]\!]_{\rho_0} \mid \gamma \in |\mathcal{D}|_n\} = \{\{(\{\alpha\}, \alpha)\} \mid \text{rg}(\alpha) < n\} \cup \{\emptyset\}$.

Let us notice that for all $n \in \mathbf{N}$ and $d \in \mathcal{D}$:
$p_{n+1} \bullet_2 d = \{\alpha \mid \alpha \in d, \text{rg}(\alpha) \leq n\} = d \cap |\mathcal{D}|_n$ and $p_0 = \{\emptyset\}$.
Thus $p_0 \bullet_2 d = \emptyset$ and $p_1 \bullet_2 d = \{\emptyset\}$ iff $\emptyset \in d$.

Now we can proof that, for all $n \in \mathbf{N}$, p_n is a projection. We must check the two points of Definition 7 *i)*. The first point is obvious, by the definition of p_n above. The second one is given by:

if $n > 0$:
$$\begin{aligned} p_n &= \{\{(a, \alpha)\} \mid \alpha \in a \cap |\mathcal{D}|_{n-1}, a \text{ is minimal}\} \cup \{\emptyset\} \\ &= \{\{(a, \alpha)\} \mid \alpha \in p_n \bullet_2 (p_n \bullet_2 a), a \text{ is minimal}\} \cup \{\emptyset\} \\ &= G(in_{\mathcal{D} \Rightarrow_s \mathcal{D}}(\Lambda(d \in \mathcal{D} \mapsto p_n \bullet_2 (p_n \bullet_2 d)))), \end{aligned}$$
and $p_0 = \{\emptyset\} = G(in_{\mathcal{D} \Rightarrow_s \mathcal{D}}(\Lambda(d \in \mathcal{D} \mapsto \emptyset))) = G(in_{\mathcal{D} \Rightarrow_s \mathcal{D}}(\Lambda(d \in \mathcal{D} \mapsto p_0 \bullet_2 (p_0 \bullet_2 d))))$.

By the definition of p_n above, the sequence $(p_n)_{n \geq 0}$ is increasing in $\mathcal{D}$ and $\bigsqcup_n p_n =$

$G(in_{\mathcal{D}\Rightarrow_s\mathcal{D}}(\Lambda(d \in \mathcal{D} \mapsto d)))$ and $p_0 \bullet_2 d = \emptyset$ for every $d \in \mathcal{D}$. Moreover, for all $n \in \mathbf{N}$ and $d_1, d_2 \in \mathcal{D}$:

i) by points 2 and 3 above $p_1 \bullet_2 d_1 \bullet_2 d_2 = \{\emptyset\} \bullet_2 d_2 = \emptyset = p_0 \bullet_2 (d_1 \bullet_2 (p_0 \bullet_2 d_2))$,

ii) if $n > 0$,

$$\begin{aligned} p_{n+1} \bullet_2 d_1 \bullet_2 d_2 &= \{\alpha \mid \exists a \subseteq d_2. \{\{(a,\alpha)\}, \emptyset\} \subseteq d_1 \cap |\mathcal{D}|_n\} \\ &= \{\alpha \mid \exists a \subseteq d_2 \cap |\mathcal{D}|_{n-1}. \{\{(a,\alpha)\}, \emptyset\} \subseteq d_1\} \cap |\mathcal{D}|_{n-1} \\ &= p_n \bullet_2 (d_1 \bullet_2 (p_n \bullet_2 d_2)). \end{aligned}$$

So $\mathcal{M}_2$ is stratified. (The proof for $\mathcal{M}_1$ differs just in the behaviour of $\bullet_1$.)

Now we must prove that $\mathcal{M}_1$ and $\mathcal{M}_2$ satisfy Definition 8 *ii)* and *iii)*. By definition of $\psi = G \circ in_{\mathcal{D}\Rightarrow_s\mathcal{D}}$, $\emptyset \in \psi(\Lambda(d \in \mathcal{D} \mapsto \emptyset))$. Moreover, by definition of $\mathcal{V}$, $\emptyset \notin \mathcal{V}$.
□

Proposition 20 *$\mathcal{M}_1 \models \eta^-$ while $\mathcal{M}_2 \not\models \eta^-$.*

4.2 Lazy stable type assignment systems

In this section we define, for each one of the previous defined models, a type system, for reasoning in a finitary way on the interpretation of terms. Namely types are names for points of the models, and a type can be assigned to a term if and only if the corresponding point is equal to (or less than) the interpretation of the term itself. As will be shown in the following, this finitary representation of the behaviour of the interpretation function, together with the approximation theorem, will be very useful for studying the local structure of the models.

Definition 21 *The set $\mathcal{T}$ of* types *is defined inductively starting from a type constant ν, in the following way:*

- $\nu \in \mathcal{T}$,
- $[\sigma_1, \ldots, \sigma_n] \to \sigma \in \mathcal{T}$ *if* $\sigma, \sigma_1, \ldots, \sigma_n \in \mathcal{T}$ *and* $(\mathtt{comp}(\sigma_i, \sigma_j))_{(1 \leq i,j \leq n)}$,

where $n \geq 0$ and comp *is formalized as in Figure 1 (see Honsell & Ronchi della Rocca (1990)).*

Types are considered modulo the equivalence:
$[\sigma_1, \ldots, \sigma_n] \to \sigma \sim [\sigma_{s(1)}, \ldots, \sigma_{s(n)}] \to \sigma$, *for every permutation s of $\{1, \ldots, n\}$.*

Definition 22 *Let $t : \mathcal{T}/\!\sim\, \to |\mathcal{D}|$ be defined as:*

- $t(\nu) = \emptyset$ *and* $t([\sigma_1, \ldots, \sigma_n] \to \sigma) = \{(\{t(\sigma_1), \ldots, t(\sigma_n)\}, t(\sigma))\}$ $(n \geq 0)$.

Property 23 *i) t is injective and surjective.*
ii) $\mathtt{comp}(\sigma, \sigma') \iff comp_{\mathcal{D}}(t(\sigma), t(\sigma'))$.

Definition 24 *i) Let a basis be a set of assumptions of the shape $x : \sigma$, where $x \in Var$ and $\sigma \in \mathcal{T}$. The* domain *of a basis B of the form $\{x_1 : \sigma_1, \ldots, x_n : \sigma_n\}$*

Judgements: $\mathtt{comp}(\sigma, \sigma')$ $\quad$ $\mathtt{nonc}(\sigma, \sigma')$
where $\sigma, \sigma' \in \mathcal{T}$. The first judgement states the compatibility, while the second states the non compatibility.

Rules:

$$\frac{}{\mathtt{comp}(\nu, \sigma)}(1) \qquad \frac{\sigma_1 \sim \sigma_2}{\mathtt{comp}(\sigma_1, \sigma_2)}(2) \qquad \frac{\mathtt{comp}(\sigma, \sigma')}{\mathtt{comp}(\sigma', \sigma)}(3) \qquad \frac{\mathtt{nonc}(\sigma, \sigma')}{\mathtt{nonc}(\sigma', \sigma)}(4)$$

$$\frac{(\mathtt{comp}(\sigma_i, \sigma'_j))_{\forall i,j \in \{1,\ldots,m\}} \quad \sigma \sim \sigma' \quad [\sigma_1 \ldots \sigma_n] \to \sigma \not\sim [\sigma'_1 \ldots \sigma'_m] \to \sigma'}{\mathtt{nonc}([\sigma_1 \ldots \sigma_n] \to \sigma, [\sigma'_1 \ldots \sigma'_m] \to \sigma')}(5)$$

$$\frac{(\mathtt{comp}(\sigma_i, \sigma'_j))_{\forall i \in \{1,\ldots,n\}, \forall j \in \{1,\ldots,m\}} \quad \mathtt{nonc}(\sigma, \sigma')}{\mathtt{nonc}([\sigma_1 \ldots \sigma_n] \to \sigma, [\sigma'_1 \ldots \sigma'_m] \to \sigma')}(6)$$

$$\frac{\exists i, j \in \{1, \ldots, \max\{n, m\}\} \,.\, \mathtt{nonc}(\sigma_i, \sigma'_j)}{\mathtt{comp}([\sigma_1 \ldots \sigma_n] \to \sigma, [\sigma'_1 \ldots \sigma'_m] \to \sigma')}(7)$$

$$\frac{(\mathtt{comp}(\sigma_i, \sigma'_j))_{\forall i \in \{1,\ldots,n\}, \forall j \in \{1,\ldots,m\}} \quad \mathtt{comp}(\sigma, \sigma') \quad \sigma \not\sim \sigma'}{\mathtt{comp}([\sigma_1 \ldots \sigma_n] \to \sigma, [\sigma'_1 \ldots \sigma'_m] \to \sigma')}(8)$$

Figure 1 Formalizing the compatibility relation

is the set dom$(B) = \{x_1, \ldots, x_n\}$, *and for each* $x \in$ *dom*(B), $B(x)$ *is defined as* $\{\sigma \mid x : \sigma \in B\}$.

ii) A consistent basis *is a basis* B *such that for each* $\{x : \sigma, x : \sigma'\} \subseteq B$, $\mathtt{comp}(\sigma, \sigma')$.

iii) The two type assignment systems prove judgements of the shapes:
$B \vdash_1 M : \sigma \qquad\qquad B \vdash_2 M : \sigma$
where B *is a consistent basis,* $M \in \Lambda$ *and* $\sigma \in \mathcal{T}$. *The rules are the following:* $(i \in \{1, 2\})$

$$\frac{}{\{x : \sigma\} \vdash_i x : \sigma}(ax) \qquad \frac{}{\vdash_i \lambda x.M : \nu}(lazy)$$

$$\frac{B \cup \{x : \sigma_1, \ldots, x : \sigma_n\} \vdash_i M : \tau \quad n \geq 0 \quad x \notin dom(B)}{B \vdash_i \lambda x.M : [\sigma_1, \ldots, \sigma_n] \to \tau}(\to I)$$

$$\frac{\begin{array}{c} B \vdash_1 M : [\sigma_1, \ldots, \sigma_n] \to \tau \quad (B_i \vdash_1 N : \sigma_i)_{1 \leq i \leq n} \\ B \cup (\bigcup_{1 \leq i \leq n} B_i) \textit{ is a consistent basis} \quad n \geq 0 \end{array}}{B \cup (\bigcup_{1 \leq i \leq n} B_i) \vdash_1 MN : \tau}(\to E_1)$$

$$\frac{\begin{array}{c} B' \vdash_2 M : \nu \\ B \vdash_2 M : [\sigma_1, \ldots, \sigma_n] \to \tau \quad (B_i \vdash_2 N : \sigma_i)_{1 \leq i \leq n} \\ B' \cup B \cup (\bigcup_{1 \leq i \leq n} B_i) \textit{ is a consistent basis} \quad n \geq 0 \end{array}}{B' \cup B \cup (\bigcup_{1 \leq i \leq n} B_i) \vdash_2 MN : \tau} (\to E_2)$$

(note that the rules of the two systems coincide, but the rule $(\to E)$.

Let $\vdash_i M : \sigma$ be an abbreviation for $\emptyset \vdash_i M : \sigma$,

Definition 25 *Let us denote, for all environments ρ, ρ',*
$dom(\rho) = \{x \in Var \mid \rho(x) \neq \emptyset\}$ and $\rho \sqsubseteq \rho' \Leftrightarrow (\forall x \in Var.\rho(x) \subseteq \rho'(x))$
and $\rho \sqsubset \rho' \Leftrightarrow (\rho \sqsubseteq \rho'$ and $\rho \neq \rho')$.
We say that two environments ρ, ρ' are compatible *if $\forall x \in Var$, $\rho(x) \cup \rho'(x) \in \mathcal{D}$. Moreover, we define $\rho \sqcap \rho'$ as the environment such that for all $x \in Var$, $(\rho \sqcap \rho')(x) = \rho(x) \cap \rho'(x)$.*

Lemma 26 *Let γ be a type, M be a term and ρ, ρ' be two compatible environments. If $t(\gamma) \in [\![M]\!]_\rho$ and $t(\gamma) \in [\![M]\!]_{\rho'}$, then $t(\gamma) \in [\![M]\!]_{\rho \sqcap \rho'}$.*

Proof. By a straightforward induction on M using stability. □

The following theorem states the soundness and completeness of the above type assignment systems w.r.t. the two models $\mathcal{M}_1$ and $\mathcal{M}_2$.

Theorem 27 *i) (Soundness) Let B be a consistent basis, $M \in \Lambda$ and $\sigma \in \mathcal{T}$. If $B \vdash_i M : \sigma$, then $t(\sigma) \in [\![M]\!]^i_{\rho_B}$, where $\rho_B(x) = \{t(\sigma) \mid \sigma \in B(x)\}$. Moreover ρ_B is minimal among the environments $\rho \in Env$ such that $\sigma \in [\![M]\!]^i_\rho$, $(i \in \{1,2\})$.*
ii) (Completeness) Given $M \in \Lambda$ and an environment $\rho \in Env$, $t(\sigma) \in [\![M]\!]^i_\rho$ implies that there is a consistent basis B such that $B \vdash_i M : \sigma$ and $B \subseteq B_\rho$, where $B_\rho(x) = \{x : \sigma \mid t(\sigma) \in \rho(x)\}$, $(i \in \{1,2\})$.

Proof. (Soundness) By induction on the derivation of $B \vdash_i M : \sigma$. We will just give the proof for the system $\vdash_2$ in case the last applied rule is $(\to E)$. The other cases are simpler.

Let $M \equiv PQ$ for some P and Q and let the basis $B \equiv B' \cup B'' \cup \bigcup_{i \in I} B_i$ for some B', B'', B_i's where $I = \{1, \ldots, n\}$. Moreover, $B' \vdash P : \nu$, $B'' \vdash P : [\sigma_1, \ldots, \sigma_n] \to \sigma$ and $B_i \vdash Q : \sigma_i$ for all $i \in I$. By induction we have:
$\emptyset \in [\![P]\!]_{\rho_{B'}}$ and $\rho_{B'}$ minimal,
$\{(a, t(\sigma))\} \in [\![P]\!]_{\rho_{B''}}$ and $\rho_{B''}$ minimal, where $a = \{t(\sigma_i) \mid i \in I\}$,
and $t(\sigma_i) \in [\![Q]\!]_{\rho_{B_i}}$ and ρ_{B_i} minimal, for all $i \in I$.
By monotonicity $a \subseteq [\![Q]\!]_{\rho_B}$, $\{(a, t(\sigma))\} \in [\![P]\!]_{\rho_B}$ and $\emptyset \in [\![P]\!]_{\rho_B}$. So, $t(\sigma) \in [\![PQ]\!]_{\rho_B}$. Moreover, ρ_B is minimal, indeed if $t(\sigma) \in [\![PQ]\!]_{\rho'}$ with $\rho' \sqsubset \rho_B$, we have $b = \{t(\gamma_1), \ldots, t(\gamma_p)\}$ for some p, such that $\{(b, t(\sigma))\}, \emptyset \in [\![P]\!]_{\rho'}$ and $t(\gamma_i) \in [\![Q]\!]_{\rho'}$ $(1 \leq i \leq p)$. By monotonicity if $\{(b, t(\sigma))\} \in [\![P]\!]_{\rho'}$ then $\{(b, t(\sigma))\} \in$

$[\![P]\!]_{\rho_B}$, and hence, by stability, $a = b$. Since $\rho_{B'}, \rho_{B''}, \rho_{B_i}, \rho' \sqsubseteq \rho_B$ then all these environments are compatible each other. By Lemma 26, $\emptyset \in [\![P]\!]_{\rho_{B'} \cap \rho'}$, $\{(a, t(\sigma))\} \in [\![P]\!]_{\rho_{B''} \cap \rho'}$ and for all i, $t(\gamma_i) \in [\![Q]\!]_{\rho_{B_i} \cap \rho'}$. Since $\rho' \sqsubset \rho_B$ we have: (i) $\rho_{B'} \cap \rho' \sqsubset \rho_{B'}$ or (ii) $\rho_{B''} \cap \rho' \sqsubset \rho_{B''}$ or (iii) $\rho_{B_i} \cap \rho' \sqsubset \rho_{B_i}$. By minimality of $\rho_{B'}$, $\rho_{B''}$ and ρ_{B_i} we have a contradiction.

(Completeness)

By induction on M. We will make the proof only for the model $\mathcal{M}_2$ because for the other model the proof is similar, but simpler.

$M \equiv x$. $t(\sigma) \in [\![x]\!]_\rho$ means $t(\sigma) \in \rho(x)$. Taking $B = \{x : \sigma\}$ we have $B \subseteq B_\rho$ and $B \vdash x : \sigma$.

$M \equiv PQ$. Since $[\![PQ]\!]_\rho = out^2_{\mathcal{D} \Rightarrow_s \mathcal{D}}(F([\![P]\!]_\rho))([\![Q]\!]_\rho)$, $t(\sigma) \in [\![PQ]\!]_\rho$ implies (by definition of $out^2_{\mathcal{D} \Rightarrow_s \mathcal{D}}$) $\emptyset \in [\![P]\!]_\rho$, $\{(a, t(\sigma))\} \in [\![P]\!]_\rho$ and $a \subseteq [\![Q]\!]_\rho$ for some a. Take $a = \{t(\sigma_1), \ldots, t(\sigma_n)\}$. By induction, $B' \vdash P : \nu$, $B'' \vdash P : [\sigma_1, \ldots, \sigma_n] \to \sigma$, $B_i \vdash Q : \sigma_i$ for all $i \in \{1, \ldots, n\}$, with $B', B'', B_i \subseteq B_\rho$. Using Rule *($\to E$)* we have $B' \cup B'' \cup \bigcup_i B_i \vdash M : \sigma$ and $B' \cup B'' \cup \bigcup_i B_i \subseteq B_\rho$.

$M \equiv \lambda x.Q$ $t(\sigma) \in [\![\lambda x.Q]\!]_\rho$ implies two cases:

1. $\sigma \sim \nu$. Then $\vdash M : \nu$, by Rule *(lazy)*.
2. $\sigma \sim [\sigma_1, \ldots, \sigma_n] \to \tau$ with $n \geq 0$.
   Since $[\![\lambda x.Q]\!]_\rho = \{\{(b, \alpha)\} \mid b \text{ minimal s.t. } \alpha \in [\![Q]\!]_{\rho[b/x]}\} \cup \{\emptyset\}$, $t(\sigma) \in [\![\lambda x.Q]\!]_\rho$ implies $t(\tau) \in [\![Q]\!]_{\rho[a/x]}$ with $a = \{t(\sigma_1), \ldots, t(\sigma_n)\}$ ($a = \emptyset$ if $n = 0$). By induction, $B \vdash Q : \tau$ with $B \subseteq B_{\rho[a/x]}$. Since $B = B_1 \cup \{x : \sigma_1, \ldots, x : \sigma_n\}$ with $x \notin dom(B_1)$, for some B_1, using Rule *($\to I$)*, $B_1 \vdash M : [\sigma_1, \ldots, \sigma_n] \to \tau$ and $B_1 \subseteq B_\rho$.

□

In the types setting the set of semantics values $\mathcal{V}$ is the set of all closed terms having a type ν. In fact it is easy to verify that $M \in \Lambda^o$ and $M \Downarrow_{lazy}$ implies $\vdash_i M : \nu$ ($i \in \{1, 2\}$).

The Approximation Theorem can be rewritten in the types setting in the following way:

Let $M \in \Lambda$, B be a consistent basis and σ be a type.

$B \vdash_i M : \sigma \iff \exists A \in \mathcal{A}(M). B \vdash_i A : \sigma$. ($i \in \{1, 2\}$).

4.3 Local structure of $\mathcal{M}_1$ and $\mathcal{M}_2$

In this subsection we give a complete characterization of the local structure of the model $\mathcal{M}_1$. Moreover we will give some properties of the local structure of $\mathcal{M}_2$. Both the approximation theorem and the type system describing the interpretation of terms are useful tools for proving these properties.

First, by Proposition 20 and Theorem 15, we know that:

Fact 28 *For each* $M, N \in \Lambda$, $M \sqsubseteq_{LS} N \Longrightarrow \mathcal{M}_1 \models M \sqsubseteq N$.

The order relation $\sqsubseteq_{LS}$ can be easily extended to approximants just posing $\Omega \sqsubseteq_{LS} A$, for every A.

Remark 29 For each $A, A' \in \mathcal{N}_{\beta\Omega}$, if $A \sqsubseteq_{LS} A'$ and $A' \sqsubseteq_{LS} A$, then $A \equiv A'$.

Lemma 30 *Let* $M, N \in \Lambda$.
$(\forall A \in \mathcal{A}(M) \,.\, \exists A' \in \mathcal{A}(M) \,.\, A \sqsubseteq_{LS} A') \Longrightarrow M \sqsubseteq_{LS} M'$.

Proof. If N or M is unsolvable then the result follows immediately.
Otherwise, $M =_\beta \lambda x_1 \ldots x_m . y M_1 \ldots M_q$ and $N =_\beta \lambda x_1 \ldots x_n . z N_1 \ldots N_p$, then it is easy to check that, for a given l, $n = m + l$, $p = q + l$, $y \equiv z$, $\forall i \leq q . \forall A \in \mathcal{A}(M_i) . \exists A' \in \mathcal{A}(N_i) . A \sqsubseteq_{LS} A'$ and $\forall j \leq l . \forall A \in \mathcal{A}(x_{m+j}) . \exists A' \in \mathcal{A}(N_{q+j}) . A \sqsubseteq_{LS} A'$. Then the result follows by induction. □

Main Lemma 31 *Let* $A, A' \in \mathcal{N}_{\beta\Omega}$.
i) $\mathcal{M}_1 \models A \sqsubseteq A' \quad \Longleftrightarrow \quad A \sqsubseteq_{LS} A'$.
ii) $\mathcal{M}_1 \models A = A' \quad \Longrightarrow \quad A \equiv A'$.

Proof. i) The proof uses a "semantic separability" property on approximants and it is given in the appendix.
ii) Immediate from i) using Remark 29. □

We can now characterize the local structure of $\mathcal{M}_1$.

Theorem 32 *Let* $M, N \in \Lambda$.
i) $\mathcal{M}_1 \models M = N \Longleftrightarrow M \approx_{LT} N$.
ii) $\mathcal{M}_1 \models M \sqsubseteq N \Longleftrightarrow M \sqsubseteq_{LS} N$.

Proof. ii) ($\Leftarrow$) By Fact 28.
($\Rightarrow$) $\mathcal{M}_1 \models M \sqsubseteq N$
$\Rightarrow \ \forall B, \sigma . (B \vdash_1 M : \sigma \Rightarrow B \vdash_1 N : \sigma)$ by Theorem 27
$\Rightarrow \ \forall B, \sigma . ((\exists A \in \mathcal{A}(M) . B \vdash_1 A : \sigma) \Rightarrow (\exists A' \in \mathcal{A}(N) . B \vdash_1 A' : \sigma))$ by Approximation Theorem
$\Rightarrow \ \forall A \in \mathcal{A}(M) . \exists A' \in \mathcal{A}(N) . A \sqsubseteq_{LS} A'$ by the Main Lemma
$\Rightarrow \ M \sqsubseteq_{LS} N$ by Lemma 30
i) By *ii)* using the fact that $\approx_{LS} = \approx_{LT}$ (Proposition 13). □

As far as the model $\mathcal{M}_2$ is concerned, we can state the following properties:

Theorem 33 *i)* $M \sqsubseteq_{LT} N \Rightarrow (\not\Leftarrow) \mathcal{M}_2 \models M \sqsubseteq N$;
ii) $M \approx_{LT} N \Rightarrow (\not\Leftarrow) \mathcal{M}_2 \models M = N$;

iii) $M \sqsubseteq_{LS} N \not\Rightarrow \mathcal{M}_2 \models M \sqsubseteq N$;
iv) $M \sqsubseteq_{PSE} N \not\Rightarrow \mathcal{M}_2 \models M \sqsubseteq N$.

Proof. i) ($\Rightarrow$) follows from Theorem 14.
A counterexample for ($\Leftarrow$) is $\mathcal{M}_2 \models \lambda x.xx = \lambda x.x(\lambda y.xy)$ (it can be easily checked using the $\vdash_2$ type system).
ii) ($\Rightarrow$) follows from i). A counterexample for ($\Leftarrow$) is the same as in the preceding point.
iii) follows from the fact that $\mathcal{M}_2 \not\models \eta^-$. For example $x \sqsubseteq_{LS} \lambda y.xy$ while, if we consider any environment ρ such that $\emptyset \notin \rho(x)$, $\rho(x) \not\subseteq \lambda y.\rho(x)y = \{\emptyset\}$ in $\mathcal{M}_2$.
iv) Let O_∞ be any unsolvable of infinite order. Then $\mathcal{M}_2 \not\models \lambda x.x \sqsubseteq O_\infty$, while, for all term M, $M \sqsubseteq_{PSE} O_\infty$. $\Box$

5 COMPARISON WITH RELATED WORK

In this section we will recall the models of the lazy λ-calculus studied in the literature and we will compare their local structures with the models $\mathcal{M}_1$ and $\mathcal{M}_2$ defined in the present paper.

Longo model The Longo's model $\mathcal{M}_L$ defined in Longo (1983) is based on the *free PSE-algebra* (see Engeler (1981)) built over a given set A, namely the pair $D_A = (\mathcal{P}(B), \bullet)$ where $B = \bigcup_n B_n$ with $B_0 = A$ and $B_{n+1} = B_n \cup \{(b, \beta) \mid b \in \mathcal{P}_{fin}(B_n), \beta \in B_n\}$ and $d_1 \bullet d_2 = \{\alpha \mid \exists a \in \mathcal{P}_{fin}(d_2).(a, \alpha) \in d_1\}$. Defining the set V as $\mathcal{P}(B) \setminus \emptyset$, it is easy to show that $\mathcal{M}_L = (\mathcal{P}(B), \bullet, V, [\![\cdot]\!]^A)$ is a lazy regular model where ψ and ϕ are defined as: $\psi(f) = \{(a, \alpha) \mid \alpha \in f(a)\} \cup A$ and $\phi(a) = \{(b, \beta) \mid (b, \beta) \in a\}$, for each continuous function f from $\mathcal{P}(B)$ to itself and $a \in \mathcal{P}(B)$. As an immediate consequence of Theorem 32 we have that the theory of $\mathcal{M}_1$ *is* the theory of $\mathcal{M}_L$. Moreover, the local structure of $\mathcal{M}_1$ *is strictly included* in the one of $\mathcal{M}_L$, whose local structure is proved in Ong (1992) to be characterized by $\sqsubseteq_{PSE}$.

Abramsky-Ong model The Abramsky–Ong model, defined in Abramsky & Ong (1993), has been built starting from a regular model in the category **CPO**, whose objects are Scott's domains and whose morphisms are continuous functions, exploiting the canonical solution $\mathcal{D}_{AO}$ of the domain equation $D \approx (D \Rightarrow D)_\perp$, where $(\cdot)_\perp$ is the usual *lifting* functor on **CPO**. Letting in and out as the constructor and destructor of lifting and (F, G) be the resulting isomorphism pair of the above equation, it is easy to check that $\mathcal{M}_{AO} = (\mathcal{D}_{AO}, \bullet, \mathcal{V}, [\![\cdot]\!]^{AO})$ is a lazy regular model where $\mathcal{V} = \mathcal{D}_{AO} \setminus \{\perp_{\mathcal{D}_{AO}}\}$ and $\phi = out_{(\mathcal{D}_{AO} \Rightarrow_c \mathcal{D}_{AO})} \circ F$ and $\psi = G \circ in_{(\mathcal{D}_{AO} \Rightarrow_c \mathcal{D}_{AO})}$. It has been showed in Abramsky & Ong (1993) that the theory of $\mathcal{M}_{AO}$ strictly includes the theory of the Longo's $\mathcal{M}_L$ and so also the one of $\mathcal{M}_1$. We conjecture both that the theory of $\mathcal{M}_{AO}$ strictly includes the one of $\mathcal{M}_2$ and the local structure of $\mathcal{M}_{AO}$ (containing the $\sqsubseteq_{PSE}$ preorder) is different from the one of $\mathcal{M}_2$ (see Theorem 33).

APPENDIX: PROOF OF THE MAIN LEMMA

For shortening, we use small letter, like a, instead of writing $[\sigma_1, \ldots, \sigma_n]$ for given types $\sigma_1, \ldots, \sigma_n \in \mathcal{T}$. For $a \equiv [\sigma_1, \ldots, \sigma_n]$, $a \rightarrow \sigma$ denotes the type $[\sigma_1, \ldots, \sigma_n] \rightarrow \sigma$. Using this notation we can say that every type is of the shape $a_1 \rightarrow \cdots \rightarrow a_n \rightarrow \nu$ with $n \geq 0$ and for all $i \in \{1, \ldots, n\}$, $a_i \equiv [\sigma^i_1, \ldots, \sigma^i_{n_i}]$ with $n_i \geq 0$ and $\sigma^i_j \in \mathcal{T}$ for $j < n_i$, and we write $\vec{[]}^p \rightarrow \sigma$, for $p \geq 0$ and type σ, for denoting the type: $\vec{[]}^0 \rightarrow \sigma = \sigma$ if $p = 0$ and $\underbrace{[] \rightarrow \cdots \rightarrow []}_{p>0} \rightarrow \sigma$ if $p \geq 1$.

Moreover, let us write $\alpha \epsilon a$ for $\alpha = \sigma_i$ for some $i \in \{1, \ldots, n\}$,

Definition 34 *Given a type $\sigma \in \mathcal{T}$, we define $l(\sigma) \in \mathbf{N}$ as: $l(\sigma) = 0$ if $\sigma \sim \nu$ and $1 + l(\tau)$ if $\sigma \sim [\sigma_1, \ldots, \sigma_n] \rightarrow \tau$, for $n \geq 0$.*

The following fact will be used in the proof:

- let M be a term. If $\sigma \sim a_1 \rightarrow \cdots \rightarrow a_n \rightarrow \nu$ and $\tau \sim b_1 \rightarrow \cdots \rightarrow b_p \rightarrow \nu$ then $(n = 0$ or $p = 0$ or $p \neq n) \Rightarrow \mathtt{comp}(\sigma, \tau)$.

Now we are ready to sketch the proof of the main lemma:
Firstly we claim that $\vdash_1$ (denoted by $\vdash$ in this section) satifies the subformula property for approximants' derivations. Let A an approximant and let d be a proof of $B \vdash A : \sigma$. Then for every subderivation d' of d, if d' proves $B' \vdash A' : \sigma'$ then both σ' and every type in B' are subformula of either σ or a type in B. This property is easily proved by induction on derivation for approximants.

Proof. *(Part $\Longrightarrow$)* It can be proved that $A \not\sqsubseteq_{LS} A'$ implies that there is at least one typing for A which is not a typing for A'. Then, the thesis follows from the completeness of the type assignment system for $\mathcal{M}_1$. The proof is a boring, but straightforward, induction on the definition of $A \not\sqsubseteq_{LS} A'$. It is not difficult to distinguish, by the type assignment system, two terms whose shapes are $A \equiv \lambda x_1 \ldots x_n . y A_1 \ldots A_p$ and $A' \equiv \lambda x_1 \ldots x_m . z A'_1 \ldots A'_q$ with either $y \not\equiv z$ or $n > m$ or $n - p \neq m - q$. So we just deal with the most difficult case which is the following:

- $A \equiv \lambda x_1 \ldots x_n . z A_1 \ldots A_p$ and $A' \equiv \lambda x_1 \ldots x_n . z A'_1 \ldots A'_p$ and $\exists i \leq n . A_i \not\sqsubseteq_{LS} A'_i$.

By induction $\exists B, \sigma$ such that $B \vdash A_i : \sigma$ and $B \not\vdash A'_i : \sigma$. Let $\mu = \vec{[]}^{i-1} \rightarrow [\sigma] \rightarrow \vec{[]}^{p-i} \rightarrow \tau$ where $\tau \in \mathcal{T}$ such that $l(\tau) > m + s + r$ where $m = \max\{l(\eta)$; where η is any subtype of either σ or a type in B $\}$, $s = \max\{q$; q arity of an occurence of z in A'_i $\}$ and $r = max\{k$; k number of abstractions in a subterm of $A'_i\}$. Let us consider $B_1 = \{z : \mu\}$. Then (see the fact given just before this proof), $\mathtt{comp}(\mu, \alpha)$ for all $z : \alpha \in B$, hence $B \cup B_1$ is a consistent basis and $B \cup B_1 \vdash z A_1 \ldots A_p : \tau$. By n applications of Rule $(\rightarrow I)$, we have $B' \vdash A : a_1 \rightarrow \cdots \rightarrow a_n \rightarrow \tau$, where, for all $i \leq n$, $a_i \equiv [\alpha^i_1, \ldots, \alpha^i_{n_i}]$ with $n_i \geq 0$, $\forall j \leq n_i . x_i : \alpha^i_j \in B \cup B_1$. (Let notice that B' can be empty.) We want to show that this typing is not possible for A'. If, by absurd, $B' \vdash A' : a_1 \rightarrow \cdots \rightarrow a_n \rightarrow \tau$, then $B \cup B_1 \vdash z A'_1 \ldots A'_p : \tau$. Then, by Rule $(\to E)$, there are B'_p and B''_p such that: $B'_p \vdash zA'_1 \ldots A'_{p-1} : a_p \to \tau$ and $B''_p \vdash A'_p : \alpha$ for all $\alpha \epsilon a_p$.

Similarly, for all $i < p$, there are B'_i and B''_i such that: $B'_i \cup B''_i = B'_{i+1}$ and $B'_1 \vdash z : a_1 \to \cdots \to a_p \to \tau$ and $B''_i \vdash A'_i : \alpha$ for all $\alpha \epsilon a_i$.

Hence, $B'_1 \cup \bigcup_{1 \le i \le p} B''_i = B \cup B_1$. So $B'_1 = \{z : a_1 \to \cdots \to a_p \to \tau\}$. Let us show that $B'_1 \not\subset B$. Indeed, $l(\tau) > \max_{z:\alpha \in B}\{l(\alpha)\}$ and $l(a_1 \to \cdots \to a_p \to \tau) \ge l(\tau)$. Thus, $B'_1 = B_1$ and $a_1 \to \cdots \to a_p \to \tau = \mu$, which implies $B''_i \vdash A'_i : \sigma$, and since for all $j \le p$ $(j \ne i).a_j \equiv []$, we have $B''_j = \emptyset$ and $B \vdash A'_i : \sigma$ or $B \cup \{z : \mu\} \vdash A'_i : \sigma$. By induction the first case is not possible. Let us suppose that the second case is true. Then there exists $zP_1 \ldots P_s$ subterm of A'_i, a base $B' \subseteq B$, a subtype ρ of μ such that $B' \cup \{z : \mu\} \vdash zP_1 \ldots P_s : \rho$ is proved by a subderivation of $B \cup \{z : \mu\} \vdash A'_i : \sigma$ and $B' \not\vdash zP_1 \ldots P_s : \rho$ (the premise on z used for typing $zP_1 \ldots P_s$ is μ). Three cases are possible: $A'_i \equiv \lambda z_1 \ldots z_r.zP_1 \ldots P_s$ or there is $y'Q_1 \ldots Q_l$ subterm of A'_i and, for some q $(1 \le q \le l)$, $Q_q \equiv \lambda z_1 \ldots z_r.zP_1 \ldots P_s$ with the variable y' either free or bound in A'_i. We consider only the last case with y' bound. The other case is similar but easier. Then $y'Q_1 \ldots Q_l$ is a subterm of $\lambda y'.Q$, which is a subterm of A'_i, and there is a subderivation of the derivation proving $B \cup \{z : \mu\} \vdash A'_i : \sigma$, whose conclusion is $B'' \vdash \lambda y'.Q : b \to \eta$ for some b and η. So there is a type $b_1 \to \cdots \to b_l \to \eta'$ belonging to b and a type $c_1 \to \cdots \to c_r \to \rho$ belonging to b_p. By the subformula property for approximants $b \to \eta$ is a subtype either of σ or of a premise in $B \cup \{z : \mu\}$. We deduce that $l(b \to \eta) \ge l(\mu) - s + r$ which is against the hypothesis on the length of μ.

(*Part* $\Longleftarrow$) By induction following the definition of $\sqsubseteq_{LS}$ and using the Theorem 27, $\square$

REFERENCES

Abramsky, S. & Ong, C.-H. L. (1993), 'Full abstraction in the lazy lambda calculus.' *Information and Computation* **105**(2), 159–267.

Asperti, A. & Longo, G. (1991), *Categories, Types and Structures.* Foundations of Computing, The MIT Press.

Bastonero, O. (1996), *Modèle fortement stables du lambda-calcul et résultats d'incomplétude.* PhD thesis, Université Paris VII.

Bastonero, O. & Gouy., X. (To Appear), 'Strong stability and the incompleteness of the class of stable models', *Annals of Pure and Applied Logic* .

Coppo, M., Dezani-Ciancaglini, M. & Zacchi, M. (1987), 'Type theories, normal forms and D_∞ lambda models.' *Information and Control* **72**(2), 85 – 116.

Engeler, E. (1981), 'Algebras and combinators.' *Algebra Universalis* **13**, 389 – 392.

Girard, J.Y., Lafont, Y. & Taylor, P. (1989), *Proofs and Types.* Cambridge University Press.

Hindley, J. R. & Longo, G. (1980), 'Lambda-calculus models and extensionality.' *Z. Math. Logik Grundlag. Math.* **26**, 289 – 310.

Honsell, F. & Lenisa, M. (1993), Some results on the full-abstraction problem for restricted λ-calculi, *in* Proceedings of 18th. Symposium on MFCS'93, 'LNCS 711', Poland, Springer-Verlag, pp. 84 – 104.

Honsell, F. & Ronchi della Rocca, S. (1990), Reasoning about interpretations in qualitative lambda models, *in* M. Broy & C. Jones, eds, 'IFIP Working Group 2.2/2.3 Working Conference on Programming Concepts and Methods', Sea of Galilee, Israel, North-Holland, pp. 505–521.

Hudak, P., Jones, S. P. & Wadler, P. (1992), 'Report on the programming language haskell.' *SIGPLAN Notices* **27**(5).

Hyland, Y. (1976), 'A syntactic characterization of the equality in some models of the lambda calculus.' *Journal of London Mathematical Society* **12**(2), 83 – 95.

Lévy, J.-J. (1975), An algebraic interpretation of the $\lambda\beta$K-calculus and a labelled λ-calculus, *in* G. Goos & J. Hartmanis, eds, 'λ-Calculus and Computer Science Theory, Lecture Notes in Computer Science 37', Springer-Verlag, Berlin, DE, pp. 147–165.

Longo, G. (1983), 'Set-theoretical models of lambda calculus: Theories, expansions and isomophisms.' *Annals of Pure and Applied Logic* **24**, 153–188.

Ong, C.-H. L. (1992), Lazy lambda calculus: Theories, models and local structure characterization, *in* W. Kuich, ed., 'Automata, Languages and Programming, LNCS 623', Springer Verlag.

Pravato, A., Ronchi della Rocca, S. & Roversi, L. (1995), Categorical semantics of the call-by-value lambda-calculus, *in* 'Proceeding of the Second International Conference of Typed Lambda Calculi and Application, TLCA'95, LNCS 902', Edinburgh, U.K., Springer-Verlag, pp. 302 –318.

Wadsworth, C. P. (1978), 'Approximate reductions and λ-calculus models.' *SIAM Journal of Computing* **7**(3), 337 – 356.

BIOGRAPHY

Olivier Bastonero was born in Paris (France) in 1966. Master in mathematics at the University Paris VII in 1990. Ph.D in Mathematics at the University of Paris VII in December 1996.

Alberto Pravato was Born in Venizia (Italia) in 1967. Master in Computer Science at the University of Udine (Italia) in the 1991. Ph.D in Computer Science at the University of Turin (Italia) in 1997.

Simona Ronchi della Rocca was born in Bra (Italia) in 1946. Master in Physics at the University of Torino in 1969. Full Professor of "Foundation of Computer Science" at the University of Torino since 1987.